Controlling the Age of Information: Buffer Size, Deadline, and Packet Replacement

Clement Kam*, Sastry Kompella*, Gam D. Nguyen*, Jeffrey E. Wieselthier† and Anthony Ephremides‡

*Information Technology Division, Naval Research Laboratory, Washington, DC
†Wieselthier Research, Silver Spring, MD
‡Electrical and Computer Engineering Department, University of Maryland, College Park, MD

***Abstract*—We study the *age of information*, which is a recently introduced metric for measuring the freshness of a continually updated piece of information as observed at a remote monitor. The age of information metric has been studied for a variety of different queuing systems, and in this work, we consider the impact of buffer sizes, packet deadlines, and packet replacement on the average age of information for queuing systems. We conduct a simulation-based study in which we modeled a wide variety of queuing systems and control mechanisms in simulation and computed the average age of information. We first study the buffer size alone to see how it affects the average age, and then we look at adding a packet deadline for such a system. We consider packet deadline control in the buffer only and in both the buffer and server, and we also compare the performance with a random deadline. We observe how the buffer size and deadline are optimized for the age, and we identify general trends for how to choose values of control mechanisms under different conditions of the packet generation rate. Lastly, we study the ability to replace packets in the buffer with newly arriving packets, and we are particularly interested in whether we can achieve the performance of such a system by controlling buffer size and deadline alone, for systems in which we do not have the ability to do packet replacement.**

I. Introduction

There are many military applications for which the timeliness or the freshness of information observed at some monitor is most pertinent, such as various tracking (e.g., target or blue force tracking) and sensing (e.g., surveillance or spectrum sensing) applications, or even networking protocols (e.g., routing). We focus on systems that are characterized by a source that transmits some status (e.g., sensor data, list of neighboring nodes) to a monitor such that the information observed at the monitor at any time was recently generated. A metric called the *age of information* or *status age* was recently proposed to study the performance of such systems with the specific goal of characterizing the freshness of information at the monitor. The *age* at the time of observation is defined as the current (observation) time minus the time at which the observed state was generated [1]–[3].

The existing research on the age metric has studied a variety of queuing systems, with different arrival/departure processes, number of servers, and queue capacities. In [1] it was shown that deterministic arrival and departure processes achieve a lower average age than memoryless processes. In [4], [5], the average age was shown to decrease as the number of servers increases. Also, it was shown in [6] that the age with a buffer capacity of zero or one can be much lower than an infinite capacity queue, and they showed that the ability to replace packets in the buffer when newer packets arrive does even better.

We are interested in further exploring the various mechanisms for controlling the age through a queue. One mechanism to study is the size of the buffer, which has only been studied for 0, 1, and infinity. It was shown in [6] that a smaller buffer size is not always better, since for some smaller values of packet arrival rate λ, the average age is lower for a buffer size of 1 than 0, but for larger λ, it is better to have a buffer size of 0. Thus, it would be interesting to investigate other buffer sizes and see which is optimal under different conditions.

Another mechanism we are interested in studying is the use of a packet deadline to discard packets in the buffer. Our prior work [7] studies the impact of this type of packet deadline for a buffer size of 1, and a properly chosen deadline was shown to improve the average age compared to not using a deadline. Further study into the age with a packet deadline for various buffer sizes as well as random deadlines, packet control in the server, and packet replacement are of interest.

In this work, we conduct a simulation study on the aforementioned control mechanisms and their impact on the average age in a single server queue. We use MATLAB to simulate a queuing system with varying buffer sizes, deadline policies, and packet replacement policies. Simulations are run for an average of 10^4 samples and results are averaged over 100 runs.

II. System Model

We study a system in which a source transmits packets to a monitor through an M/M/1/k queue, which has a total capacity of $k-1$ packets in the queue and one packet in service. Typically, an arriving packet that encounters a full capacity system never enters the system and is dropped. A system may or may not have a deadline imposed on packets in the system, in which a packet that is waiting in queue for a time period longer than the deadline is dropped. We consider two cases, where a packet can and cannot be dropped due to an expired deadline after entering the server. A plot of the age of information for an M/M/1/2 system with a packet deadline is shown in Figure 1, where transmissions occur at times $t_1, t_2, \ldots$, and receptions at the monitor occur at times $t'_1, t'_2, \ldots$.

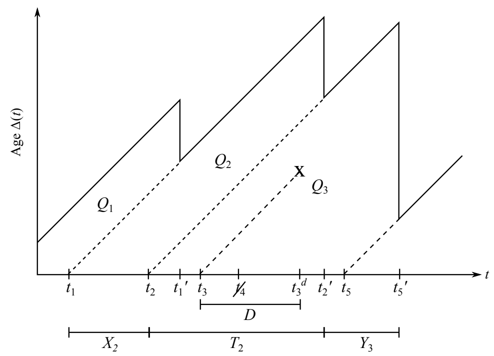

Fig. 1. Age of information for an M/M/1/2 system with a packet deadline.

We refer to the time between packet generations as the interarrival time $X_i, i = 2, 3, \ldots$, which is equal to $t_i - t_{i-1}$. The interarrival times are modeled as random; consequently, the source does not have control over the exact times at which it can transmit updates. In our model, the X_i's are i.i.d. exponential random variables with rate λ.

We call the time spent in the server by packet k the service time $S_k, k = 1, 2, \ldots$, which is equal to $t_k' - t_k$. The service time S_k is modeled as exponential with rate μ, and all the S_k's are i.i.d. and independent of the X_i's. The total time spent in the system from arrival to service is given by $T_k, k = 1, 2, \ldots$, where $T_k = W_k + S_k$, with W_k being the time spent waiting in the queue.

The *age of information* at time t is defined as $\Delta(t) = t - u(t)$ [1], where $u(t)$ is the timestamp of the most recent information at the receiver as of time t. Given this definition, we can see that the age increases linearly with t but is reset to a smaller value with each packet received that contains newer information, resulting in the sawtooth pattern shown in Figure 1.

We also define the *interdeparture time* Y_k as the time between the instants of complete service for the $k-1$st packet served and the kth packet served. This will be useful in the computation of the average age.

III. Effect of Buffer Size

We first consider the impact of the buffer size on the age of information. This was studied for a buffer size of 0 and 1 in [8], which can be analyzed using a graphical argument (average value of the curve in Fig. 1) to arrive at the expression

$$\Delta = \lambda_e\Big(\frac{1}{2}E[Y_k^2] + E[T_{k-1}Y_k]\Big). \tag{1}$$

Solving for the various terms in (1), the average age for the M/M/1/1 and M/M/1/2 were shown to be

$$\Delta_{M/M/1/1} = \frac{1}{\lambda} + \frac{2}{\mu} - \frac{1}{\lambda+\mu} = \frac{1}{\mu}\Big(\frac{1}{\rho} + 2 - \frac{1}{\rho+1}\Big) \tag{2}$$

$$\begin{aligned}\Delta_{M/M/1/2} &= \frac{1}{\lambda} + \frac{3}{\mu} - \frac{2(\lambda+\mu)}{\lambda^2+\lambda\mu+\mu^2} \\ &= \frac{1}{\mu}\Big(\frac{1}{\rho} + 3 - \frac{2(\rho+1)}{\rho^2+\rho+1}\Big)\end{aligned} \tag{3}$$

From (2) and (3), we determine here which values of $\rho = \lambda/\mu$ the age $\Delta_{M/M/1/1}$ is less than $\Delta_{M/M/1/2}$ (and vice versa):

$$\begin{aligned}2 - \frac{1}{\rho+1} &< 3 - \frac{2(\rho+1)}{\rho^2+\rho+1} \\ \frac{2(\rho+1)}{\rho^2+\rho+1} - \frac{1}{\rho+1} &< 1 \\ 0 &< \rho(\rho^2+\rho-1)\end{aligned}$$

For $\rho > 0$, the inequality is true when $\rho > (-1+\sqrt{5})/2 \approx 0.618$. That is, when $\rho > 0.618$, M/M/1/1 achieves a lower age than M/M/1/2, since packets are arriving frequently enough relative to the service rate. However, when $\rho < 0.618$, M/M/1/2 achieves a lower age since packets do not arrive frequently enough, and it helps to have an update packet stored in the buffer to be transmitted.

From this analysis of the average age of the M/M/1/1 and M/M/1/2, we have demonstrated that the relationship between age and buffer size is not simple, in that while having a smaller buffer reduces the waiting time, this does not necessarily achieve a lower age. Given the complexity of solving for the average age for the M/M/1/1 and M/M/1/2 systems and the certain complexity of solving for the average age of the general M/M/1/k system, we use simulation to find the age for an M/M/1/k system, for various values of k. The results are plotted in Figure 2 for $\mu = 1$. We again see that for lower packet arrival rate λ, increasing the buffer size actually leads to a slight decrease in the average age, but for larger λ, larger buffer sizes have a more detrimental impact on the average age.

If we fix the buffer size to be 0 (M/M/1/1), the average age is strictly decreasing in λ since there is no waiting in the buffer. For buffer sizes greater than 0, the average age initially decreases in λ, but eventually starts to increase since at some point packets are arriving frequently enough so that it is not necessary to hold any in the buffer. This effect is more significant for larger buffer sizes, but even for the M/M/1/2 case, there is a point at which the average age starts increasing. We can determine that point as follows:

$$\begin{aligned}\frac{\partial \Delta_{M/M/1/2}}{\partial \rho} = -\frac{1}{\rho} - \frac{2}{\rho^2+\rho+1} + \frac{2(\rho+1)(2\rho+1)}{(\rho^2+\rho+1)^2)} &= 0 \\ \rho^4 + 2\rho^3 - 3\rho^2 - 2\rho - 1 &= 0\end{aligned}$$

For non-negative and non-complex values of ρ, the last equality is true for $\rho \approx 1.427$. In our simulations, the minimum among $\lambda = 0.25, 0.5, \ldots, 1.75, 2$ occurs at $\lambda = 1.5$. As the buffer size increases, the value of λ at which the minimum age occurs decreases, since this prevents the buffer from filling up with older packets.

The optimum average ages for $\lambda = 0.25$, 0.5, 1, and 1.5 and the optimum buffer size are provided in Table I. For $\lambda = 0.25$, a buffer size of 10 is optimal, but for $\lambda = 0.5$, the optimum buffer size quickly goes to 1, and then to 0 for larger values of λ (recall, M/M/1/1 is better than M/M/1/2 for $\rho > 0.618$). Over all values of λ and buffer sizes, the minimum age is achieved for a buffer size of 0 and $\lambda \to \infty$, which achieves an average age of $2/\mu$ (can be derived from (2)).

TABLE I
OPTIMUM AVERAGE AGE, EFFECT OF BUFFER SIZE

λ	Minimum Age	Optimum Buffer Size	% improvement vs. M/M/1/1	% improvement vs. M/M/1/2
0.25	5.0698	10	2.49%	0.51%
0.5	3.2850	1	1.28%	–
1	2.4997	0	–	6.19%
1.5	2.2705	0	–	13.13%
2	2.1682	0	–	18.01%

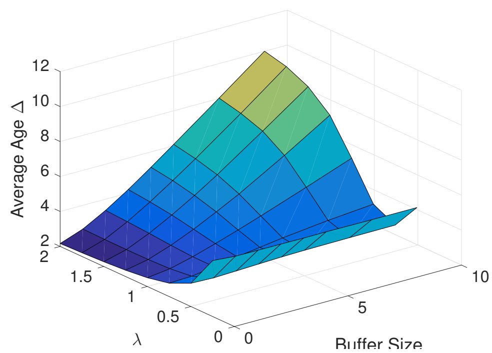

Fig. 2. M/M/1/k, $\mu = 1$.

IV. EFFECT OF DEADLINE

A. Packet Control in Buffer Only

We now add the element of subjecting each packet to a deadline, where a packet that has not been served may get dropped from the system if the deadline expires. We first consider the case where the deadline affects packets in the buffer only, i.e., packets that make it to the server before the deadline are never dropped. We mathematically analyzed the M/M/1/2 system with a deadline in [7], again solving for the terms in (1). Our numerical results showed that a properly chosen deadline can improve the age compared to the M/M/1/1 and M/M/1/2 without a deadline.

We simulated the M/M/1/k system with a deadline and provide the results in Figures 3–5 for $\lambda = 0.5$, 1, 1.5, 2, and $\mu = 1$. For smaller λ ($= 0.5$, Fig. 3), we observe that increasing the deadline reduces the age, and that larger buffer sizes seem to do better, since more packets are stored and the deadline prevents them from getting too stale in the buffer. For $\lambda = 1$ (Fig. 4), the age appears to decrease and then increase with the deadline, and a similar phenomenon is observed with the buffer size. If the deadline is too small, then packets are removed too quickly which leads to fewer updates. If a deadline is too large, packets can get too stale and it would be better to drop them and wait for another packet to arrive. For a properly chosen deadline, the buffer size should also be carefully chosen for this value of λ. The minimum average age is lower than that of the $\lambda = 0.5$ case by 24% (2.4454 vs. 3.2258) and is achieved with a buffer size of 24 and a deadline of 0.5. For a larger $\lambda = 1.5$ (Fig. 5), a lower age is achieved with a smaller deadline, since packets are generated frequently enough to be trimmed via deadline, so, in effect, fresh packets are sent at a sufficiently high rate. Of the values simulated, the optimum occurs here for a buffer size of 14 and a deadline of 0.3. The minimum average age is lower than that of the $\lambda = 0.5$ case by 31% (2.2369 vs. 3.2258). Finally, for $\lambda = 2$, the optimum buffer size is still 14 but the optimum deadline is reduced to 0.1. The minimum average age is lower than that of the $\lambda = 0.5$ case by 33% (2.1486 vs. 3.2258).

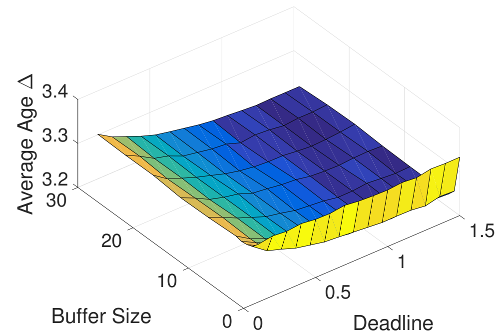

Fig. 3. M/M/1/k with deadline, packet control in buffer only, $\lambda = 0.5$, $\mu = 1$.

We have provided the optimal ages for the four values of λ in Table II. The observation is as follows: as λ increases, the minimum age decreases, and it is achieved by reducing the deadline, since more arrivals requires more aggressively trimming the packets in queue. The buffer size does not have as clear of a trend. It is noted that the % improvement appears small compared to the case with no deadline, but such an improvement may be critical for a real-time system. In addition, we will see in the next section that the deadline has a significantly greater effect when it can affect the packet in the server.

TABLE II
OPTIMUM AVERAGE AGE, PACKET CONTROL IN BUFFER ONLY

λ	Minimum Age	Optimum Buffer Size	Optimum Deadline	% improvement vs. no deadline
0.5	3.2258	19	1.2	1.80%
1	2.4454	24	0.5	2.17%
1.5	2.2369	14	0.3	1.48%
2	2.1486	14	0.1	0.90%

B. Packet Control in Buffer and Server

In this section, we also study a case of using a deadline, but now both packets in the server and the buffer can be dropped if the deadline expires. The simulation results are provided in Figures 7-9. For all λ, the age is relatively large at smaller deadlines because the packets in the server are now also subject to a deadline, so most packet are dropped. As the deadline starts increasing, the age starts to decrease. The

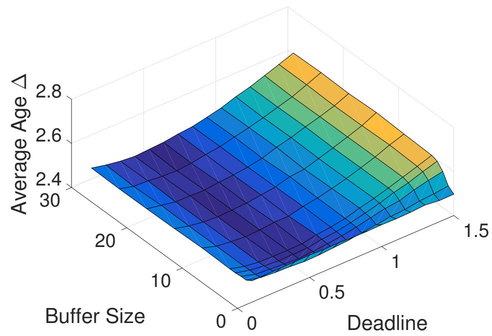

Fig. 4. M/M/1/k with deadline, packet control in buffer only, $\lambda = 1$, $\mu = 1$.

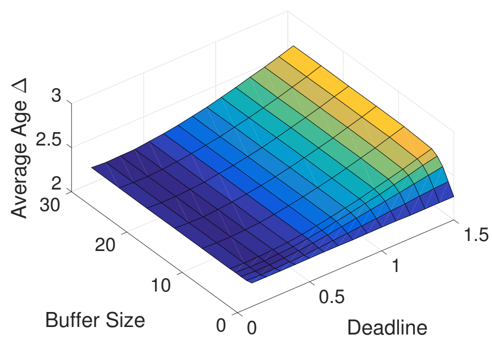

Fig. 5. M/M/1/k with deadline, packet control in buffer only, $\lambda = 1.5$, $\mu = 1$.

TABLE III
OPTIMUM AVERAGE AGE, PACKET CONTROL IN BUFFER AND SERVER

λ	Minimum Age	Optimum Buffer Size	Optimum Deadline	% improvement vs. buffer only
0.5	3.0305	24	2	6.05%
1	1.9034	14	1.6	22.16%
1.5	1.4820	14	1.2	33.75%

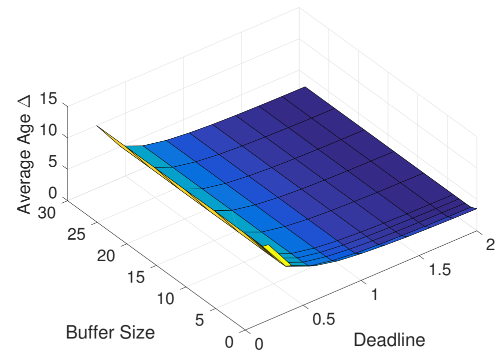

Fig. 7. M/M/1/k with deadline, packet control in buffer and server, $\lambda = 0.5$, $\mu = 1$.

age and the optimum deadline and buffer size are provided in Table III. We observe that the optimum deadline decreases as λ increases. As in the case with packet control in the buffer only, the buffer size does not show as clear of a trend. In this case, the average age is less sensitive to the buffer size once it is sufficiently large (≥ 1). Overall, we observe that the ability to drop packets in the server using a deadline can improve the age by as much as 33% when compared to only dropping packets in the buffer.

C. Random Deadline

We also consider the case of having a random deadline, specifically an exponentially distributed deadline generated for each packet, and compare with a deterministic deadline. We are interested in a model that is analytically tractable and can provide a closed-form expression, so we choose an exponentially distributed deadline, since it restores the memorylessness of the system. It would be helpful if the performance with the random deadline can approximately achieve that of the deterministic deadline. In Figure 10, we plot the average age vs. deadline for the M/M/1/2 system, where for the random deadline case, the deadline is actually the average deadline. The average age for the random case is similar to the deterministic case for smaller (average) deadline. As the deadline increases, the age in both the random and deterministic case go down, but the random case is less steep. As the deadline increases further, the age for the random case increases less than in the deterministic case. The randomly generated deadline appears to have the effect of smoothing out the average age relative to the deterministic case.

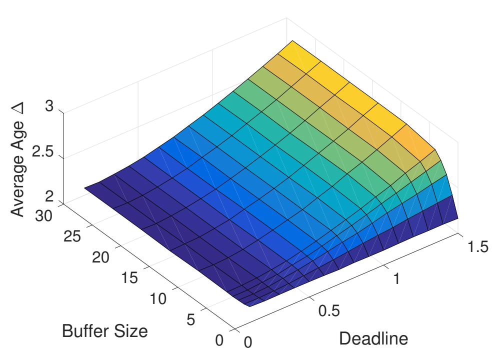

Fig. 6. M/M/1/k with deadline, packet control in buffer only, $\lambda = 2$, $\mu = 1$.

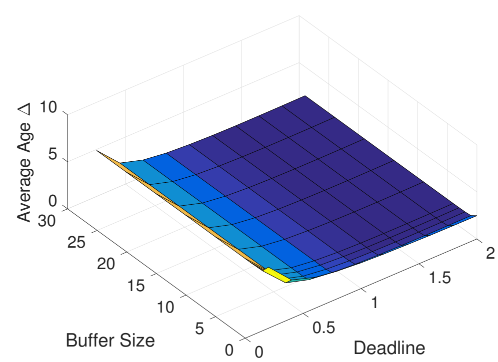

Fig. 8. M/M/1/k with deadline, packet control in buffer and server, $\lambda = 1$, $\mu = 1$.

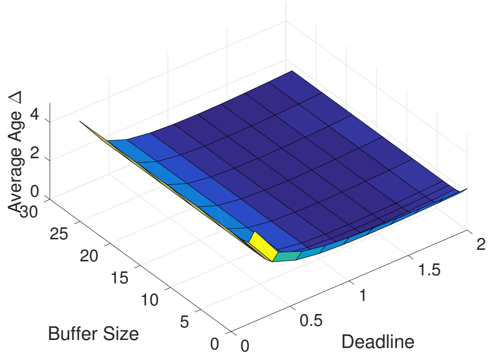

Fig. 9. M/M/1/k with deadline, packet control in buffer and server, $\lambda = 1.5$, $\mu = 1$.

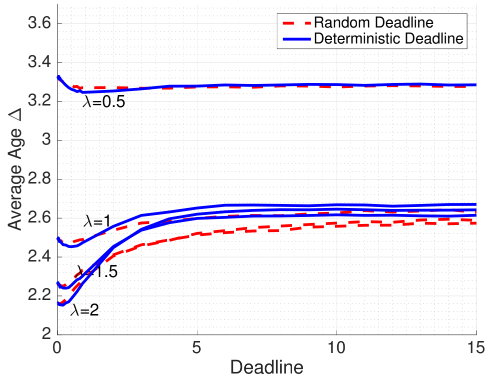

Fig. 10. M/M/1/2 with Random Deadline, $\mu = 1$.

The minimum age for the random deadline for the M/M/1/2 system and the comparison to the deterministic deadline case is given in Table IV. We observe that if the deadline is chosen properly, there is only about a 0.7% loss in using an exponentially distributed random deadline vs. a deterministic deadline, which helps assess the value of a theoretical analysis that utilizes random deadlines.

TABLE IV
OPTIMUM AVERAGE AGE FOR M/M/1/2, PACKET CONTROL IN BUFFER, RANDOM DEADLINE

λ	Minimum Age	Optimum Deadline	% loss vs. M/M/1/2, deterministic deadline
0.5	3.2676	0.8	0.65%
1	2.4711	0.5	0.73%
1.5	2.2499	0.2	0.44%
2	2.157	0.04	0.20%

V. EFFECT OF PACKET REPLACEMENT CAPABILITY

We now consider a greater level of packet control in the queue, in which we have the ability to replace an old packet in the buffer upon arrival of a new packet. In this case, no new packets are blocked from entering a full system. This was analyzed in [6] and [8] (denoted as M/M/1/2*) and was shown to achieve a lower age than the M/M/1/1 and M/M/1/2 without this packet replacement capability. If we always choose to replace a packet in the buffer with a newly arriving packet, having a buffer size larger than one has no impact since any arriving packet that sees a packet in the buffer will replace it. Therefore, our focus here is on the M/M/1/2 system with packet replacement. Other types of queue management, such as a last-come, first-served discipline or replacing packets after more than 1 packet arrives in the buffer, are outside the scope of this paper.

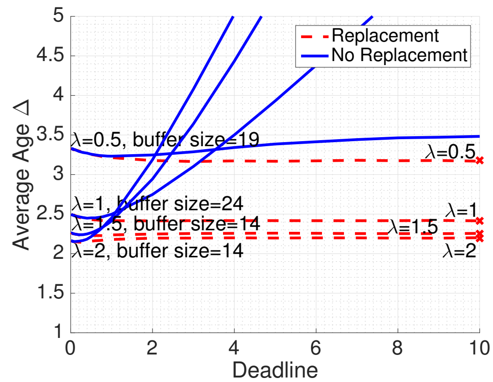

Fig. 11. M/M/1/2* with deadline.

A. Packet Control in Buffer Only

We first consider the impact of a deadline on an M/M/1/2 system with packet replacement, where a packet can only be dropped in the buffer if its deadline expires. We plot the results of our simulations in Figure 11 for $\lambda = 0.5$, 1, 1.5, 2, and $\mu = 1$. We see that the deadline has a very small impact on the average age. This suggests that the packet replacement capability prevents the deadline from having much of an impact on the average age. Of the λ plotted, only for $\lambda = 2$ does the age noticeably (albeit slightly) decrease as the deadline decreases. The minimum age for the M/M/1/2 with packet replacement is shown in Table V, and the % improvement over the case where there is no deadline is shown to be as much as 2%.

We are interested in studying whether the performance of a system with packet replacement policy can be achieved with a deadline only. Figure 11 includes the results for M/M/1/k with a deadline in the buffer for $\lambda = 0.5$, 1, 1.5, 2 and we choose the buffer size that minimizes the age according to Table II. Table V shows the % improvement of the M/M/1/2 with packet replacement over these M/M/1/k without packet replacement results, minimized over the deadline. If the buffer size and deadline are chosen properly, the age for the M/M/1/k without packet replacement can be close to that of the case with packet replacement, particularly for higher λ.

B. Packet Control in Buffer and Server

Now we consider the case where in addition to packets in the buffer, packets in the server can be dropped due an

TABLE V

OPTIMUM AVERAGE AGE WITH PACKET REPLACEMENT, PACKET CONTROL IN BUFFER ONLY

λ	Minimum Age	Optimum Deadline	% improvement vs. no deadline	% improvement vs. no packet replacement, over all deadlines
0.5	3.1652	3	0.29%	2.08%
1	2.4145	2	0.09%	1.39%
1.5	2.227	0.8	1.2%	0.53%
2	2.1441	0.3	2.47%	0.36%

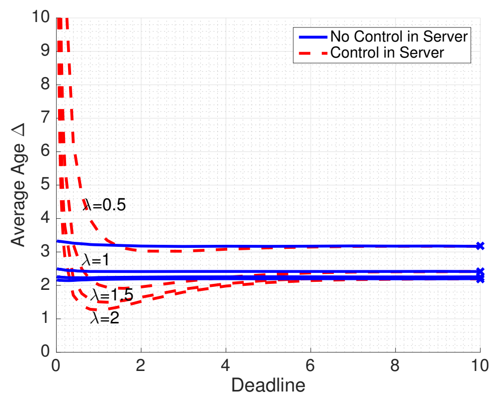

Fig. 12. M/M/1/2* with deadline and packet control in server.

expiring deadline. The results for our simulations are provided in Figure 12, where we see that the deadline has a significant effect on the average age when it affects the packet in the server. The minimum age values are provided in Table VI, where the age improvement is up to 40% ($\lambda = 2$) compared to the buffer only case. However, the age is highly sensitive to the deadline in the vicinity of the minimum age, since the age approaches infinity as the deadline approaches 0.

TABLE VI

OPTIMUM AVERAGE AGE WITH PACKET REPLACEMENT, PACKET CONTROL IN BUFFER AND SERVER

λ	Minimum Age	Optimum Deadline	% improvement vs. buffer only
0.5	3.022	3	4.52%
1	1.9125	1.6	20.79%
1.5	1.4968	1.2	32.79%
2	1.2667	1	40.92%

VI. CONCLUSION AND FUTURE WORK

Following up our theoretical work on the use of a packet deadline as a way to optimize the age of information, we conducted a simulation-based study on a broader range of control mechanisms and their effect on the age, independently and jointly. We observe that when we consider the buffer size alone, our theoretical analysis shows that the age for the M/M/1/2 system is smaller than that of the M/M/1/1 for $\rho < 0.618$. For $\rho > 0.618$, our simulations show that the age can increase quite drastically as the buffer size increases. Considering λ and buffer size jointly, the best performance occurs when the buffer is reduced to zero and λ is increased as much as possible. When we add the packet deadline as a control mechanism for packets in the buffer only, there is a slight increase in age performance (2%) if the deadline is decreased as λ increases. If the deadline can affect packets in the server, there is a major increase in the age performance (33% at $\lambda = 1.5$). If we add the ability to replace packets in the buffer, the deadline has a small effect on the age performance over all values of the deadline. The best performance for the case without packet replacement can be very similar to that of the case with packet replacement, if the buffer size and deadline are chosen optimally. When the deadline can impact the packet in the server, the improvement is again more significant as in the case without packet replacement, but the age is highly sensitive to the deadline in the vicinity of the optimal age. Choosing a slightly lower deadline can lead to a significant increase in the average age.

Future work includes confirming the simulation results in theory. We would also like to study other control mechanisms, such as packet replacement that only occurs after more packets are in the buffer; last-come, first-served queuing discipline; and allowing packet replacement on packets in the server.

ACKNOWLEDGMENT

This work was supported by the Office of Naval Research.

REFERENCES

[1] S. Kaul, R. Yates, and M. Gruteser, “Real-time status: How often should one update?” in *Proc. IEEE INFOCOM*, Orlando, FL, Mar. 2012, pp. 2731–2735.

[2] ——, “Status updates through queues,” in *Conference on Information Sciences and Systems (CISS)*, Princeton, NJ, Mar. 2012, pp. 1–6.

[3] R. D. Yates and S. Kaul, “Real-time status updating: Multiple sources,” in *Proc. IEEE International Symposium on Information Theory (ISIT)*, Cambridge, MA, Jul. 2012, pp. 2666–2670.

[4] C. Kam, S. Kompella, and A. Ephremides, “Age of information under random updates,” in *Proc. IEEE International Symposium on Information Theory (ISIT)*, Istanbul, Turkey, Jul. 2013, pp. 66–70.

[5] ——, “Effect of message transmission diversity on status age,” in *Information Theory (ISIT), 2014 IEEE International Symposium on*, June 2014, pp. 2411–2415.

[6] M. Costa, M. Codreanu, and A. Ephremides, “Age of information with packet management,” in *IEEE International Symposium on Information Theory (ISIT)*, June 2014, pp. 1583–1587.

[7] C. Kam, S. Kompella, G. D. Nguyen, J. E. Wieselthier, and A. Ephremides, “Age of information with a packet deadline,” in *Information Theory (ISIT), 2016 IEEE International Symposium on*, July 2016.

[8] M. Costa, M. Codreanu, and A. Ephremides, “On the age of information in status update systems with packet management,” *CoRR*, vol. abs/1506.08637, 2015. [Online]. Available: http://arxiv.org/abs/1506.08637